ICT Seventh Framework Programme (ICT FP7)

Grant Agreement No: 318497

Data Intensive Techniques to Boost the Real – Time Performance of Global Agricultural Data Infrastructures

D1.3.2: Annual Public Report

Deliverable Form	
Project Reference No.	ICT FP7 318497
Deliverable No.	D1.3.2
Relevant Workpackage:	WP1: Project Management
Nature:	R
Dissemination Level:	PU
Document version:	Final
Date:	5/12/2014
Authors:	UAH
Document description:	The public report of the second year summarizes project progress.

1. Project Context and Objectives

This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 318497

During the last years, the trend to open up data and provide them freely on the Internet has intensified in volume as well as quality and value of the data made available. The linked data community has grasped the opportunity to combine, cross-reference, and analyse unprecedented volumes of high-quality data and to build innovative applications. This effort has caused a tremendous network effect, adding value and creating new opportunities for everybody, including the original data providers.

But most of the low-hanging fruit has been picked and it is time to move on to the next step, combining, cross-indexing and, in general, making the best out of all public data, regardless of their size, update rate, and schema; accepting that centrally-managed repositories (even distributed) are not able to meet the challenges ahead and that we need to develop the infrastructure for the efficient querying of loose data source federations at a large scale.

SemaGrow carries out fundamental databases research and develops methods and infrastructure that will be rigorously tested on current use cases as well as on their projected data growth beyond project's end: we are laying the foundations for the scalable, efficient, and robust data services needed to take full advantage of the data-intensive and inter-disciplinary Science of 2020.

2. Use Cases

The SemaGrow project uses the large-scale and complex *agricultural data service* ecosystem as a testbed for its technologies. During the first reporting period, the project engaged consortium members and external stakeholders to identify user needs and relevant data sources and to draft requirements and the system architecture.

From the perspective of engaging stakeholders and translating user needs to requirements, the project focused on the definition of different types of use cases based on stakeholder requirements. We documented the outcomes of the stakeholder workshops, as it is from the contact with stakeholders that the project may effectively pin point how to develop and then evaluate its results. Through the definition of use cases, which is the core of this deliverable, it translates the expressed ideas and requirements to application descriptions that will form the basis for the development of service demonstrators and draws the connection between the needs of the stakeholders and the technological developments of the project.

Three categories of use cases are considered in the SemaGrow project, addressing a diverse group of stakeholders and wide area of application:

- *Heterogeneous Data Collections and Streams,* focusing on data-intensive experiments in the domain of agricultural and forestry modelling. SemaGrow technologies are used to prepare suitable input dataset for modelling experiments from the wealth of heterogeneous big data collections and streams available. The stakeholders are the *modellers* who prepare the experiments and the *IT personnel* who support them. These use cases validate the integrative semantic capabilities provided by the Semagrow Stack as well as its ability to search and retrieve large data volumes.
- *Reactive Data Analysis*, focusing on the *AGRIS portal* that serves scientific bibliography and relevant Web resources. SemaGrow technologies are used to federate the diverse endpoints used to search for and retrieve resources that are semantically relevant to a given AGRIS bibliography item. The stakeholders are the *Web developers* who maintain the portal and the *data analysis experts* who experiment to refine the relevance scoring mechanism. These use cases test validate the ease of adding members to the federation and the reactivity of the Semagrow Stack when searching through big data in order to find results that are not necessarily voluminous.

- *Reactive Resource Discovery*, focusing on the *Agricultural Discovery Space (ADS)* application that serves diverse bibliographical and educational resources over a simple NoSQL REST API. The stakeholders are the *LOD professionals and enthusiasts* who put together Web applications using Web services and the ADS end-users who are searching for resources. These uses cases validate the ease of developing Web applications over the SemaGrow Stack and the ability to federate and semantically integrate a large number of heterogeneous data sources, although the contents of any individual data source do not constitute big data by itself.

The assumption is that the efficiency of the stakeholders on these use cases will significantly improve when replacing current methods of data access with SemaGrow technologies *without affecting their workflow otherwise*. That is to say that, for example, agricultural and forestry modelling software, the AGRIS portal, and the clients developed for the ADS Web application will not need to be re-developed in order to be able to take advantage of SemaGrow technologies. The rigorous testing and evaluation activities foreseen in SemaGrow are meant to validate both the increase in stakeholder efficiency and the effort required to adopt SemaGrow as data access infrastructure.

3. Technology Development

By addressing these use cases, the project evaluates its approach to the following major technical challenges:

- *Finding small results in big data:* in this "needle in a haystack" situation we are joining results from different big datasets in order to retrieve a result set that does not constitute big data by itself. The challenge here is to have an intelligent *query execution planner* that guides the *query execution engine* along a query execution plan that never retrieves large quantities of results that do not contribute to the end-result because they do not join with subsequent query patterns. Such a query execution planner relies heavily on the availability of accurate instance-level metadata.
- *Big results from big data:* in this situation the result set constitutes big data, and no amount of query plan optimization can avoid this, since this is what has been requested by the client. Besides the challenges for the query execution engine, handling this situation is also relevant to the ability of the histogram maintenance mechanism to efficiently handle query feedback that is big data.
- *Integrating heterogeneous data sources:* accepting that some schemas might be better suited to a given dataset and application and that there is no consensus about a "universal" schema or vocabulary for any given application, we are developing technologies that allow data providers to publish in the manner and form that best suits their processes and purposes and data consumers to query in the manner and form that best suits theirs. Such technologies discover alignments that minimize losses and plan query execution so that losses do not accumulate over successive schema translations.

We proceed to present progress during the second period along these major technical objectives, as well as progress in developing the overall SemaGrow system.

3.1 Managing Large Scale

The central product of the project is the SemaGrow Stack: a deployment of the SemaGrow Stack federates remote SPARQL end-points and offers a single endpoint to its clients. During its second period, the SemaGrow project developed the *Query Decomposition* component that calculates the optimal *query execution plan* for a given query. Query execution plans break down a query into fragments and specify the federated end-point that will execute it and the optimal execution *ordering* for merging the partial results returned for each fragment. This calculation is based on the estimated *cost* of executing a query fragment on a given endpoint. Cost is estimated from *metadata* about the federated data sources: schema-level information but also instance-level statistics, served from the *Resource Discovery* component. During the project's first period, cost estimation was based on metadata expressed using the *Vocabulary of Interlinked Datasets (VoID)* and provided manually using ELEON, a specialized RDF authoring environment.

We have now extended VoID into *Sevod*, an RDF vocabulary that is expressive enough to represent detailed instance-level metadata akin to relational database *histograms*. This improves cost estimation, but *Sevod* models are practically impossible to maintain manually. To address this, we developed and prototyped a method for automatically building and maintaining *Sevod* descriptions without imposing any extra overhead queries and without requiring access to data dumps. The core idea is based on *adaptive query processing* from the relational database literature, where histograms are maintained by analysing the results to the user-requested query workload. Database histograms typically target *numerical attributes*, exploiting the fact that groupings of numerical values can be succinctly described by a range.

Research in SemaGrow extended the state-of-the-art to the domain of *strings*, exploiting the fact that there are several classes of strings that have an internal structure that can be used to express "ranges", such as prefixes over filesystem paths and URIs. This has given us very promising results, although our experiments have given us ideas for further improvements during the final project period.

Although the research above has given us considerable advances towards handling the needle-in-a-haystack problem of identifying the best strategy to retrieve data out of a large-scale repository, intelligent execution planning does not improve retrieving large amounts of data and also breaks down when remote endpoints are unexpectedly slow or unresponsive. To efficiently combine query results at a large scale and to handle the dynamic nature of the LOD cloud, we are carrying over and adapting optimized query execution methods from relational database research.

3.2 Managing Heterogeneity

Besides integrating large-scale data sources, SemaGrow also tackles integrating *heterogeneous* data sources. This pertains to efficiently applying resource mappings within the query execution engine of the SemaGrow Stack, but also to a system of tools that complement the SemaGrow Stack and that provide the mappings that should be applying. While the first period of the project focused on developing technologies that support robust alignment over heterogeneous knowledge models and languages (MAPLE, Lime) and on refining the SYNTHESIS system for automatic alignment, the second period focused on robustness and integration. Specifically, SYNTHESIS has been provided with an appropriate data access layer, moved to a thread-based architecture, and provided with ontology modularization methods to improve scalability. Furthermore, the architecture of the VocBech environment for manual alignment and of its backing framework, Semantic Turkey, has been reworked to allow for multi-project and multi-graph management, dynamic context injection inside the services, seamless navigation of local and web data laying out a common ground for a user-centric ontology alignment experience. Semantic Turkey also features a new Linked Data explorer that will be integrated into VocBench during the final period of the project.

Finally, during its second period the project experimented with the CODA system for knowledge extraction and transformation, applying it to a datasheet import scenario. The objective is to strike a balance between enforcing conventions (affording ease of use) and capacity to deal with complexity. This effort resulted in Sheet2RDF, a *datasheet to RDF* import and transformation system, which by following a "convention over configuration" approach makes so that the "evident" and trivial imports can be dealt in an almost automatic way, whereas more complex transformations can be still managed through the more powerful capabilities of CODA.

3.3 Experimentation and Testing

During the second period, the project has worked to develop the experimental setups needed in order to evaluate the technical efficiency of the SemaGrow Stack. The aim is to develop setups that will not only be used to technically evaluate project outcomes, but can also be implemented as an automated testing component that monitors system health in SemaGrow Stack deployments.

We established and ran the following experiments on query execution and managing large scale:

- The AGRIS database dumps are distributed per year of addition to the AGRIS database. Using this, we were able to set up an experiment that tests the convergence of our self-tuning histograms to database updates. Besides evaluating the accuracy of our self-tuning histograms method, this experiment will also be the basis of an automated testing component that periodically tests the accuracy of the federated endpoint metadata. A particularly important run-time question is the relation between the space allocated for histograms and their accuracy. This question cannot be answered in advance but only dynamically, since it is affected by the workload applied to each deployment.
- We have measured the optimality of the query execution plans calculated by our Query Decomposition component and the efficiency of our query execution engine using the FedBench benchmark to compare SemaGrow against FedX and SPLENDID. This experiment is done for evaluation purposes only and no automated testing component will be derived.
- Based on the FedBench experiments, on-going work is to set up experiments over the integrated histogram maintenance/query execution system in order to measure the impact of accurate histograms on actual planning decisions. This will allow the automated testing component to estimate the impact that reducing or increasing histogram size will have to (a) the optimality of querying plans and (b) the overheads imposed by histogram maintenance and accessing. Such decisions are critical in deployments over large datasets, as the histogram itself can become big data.

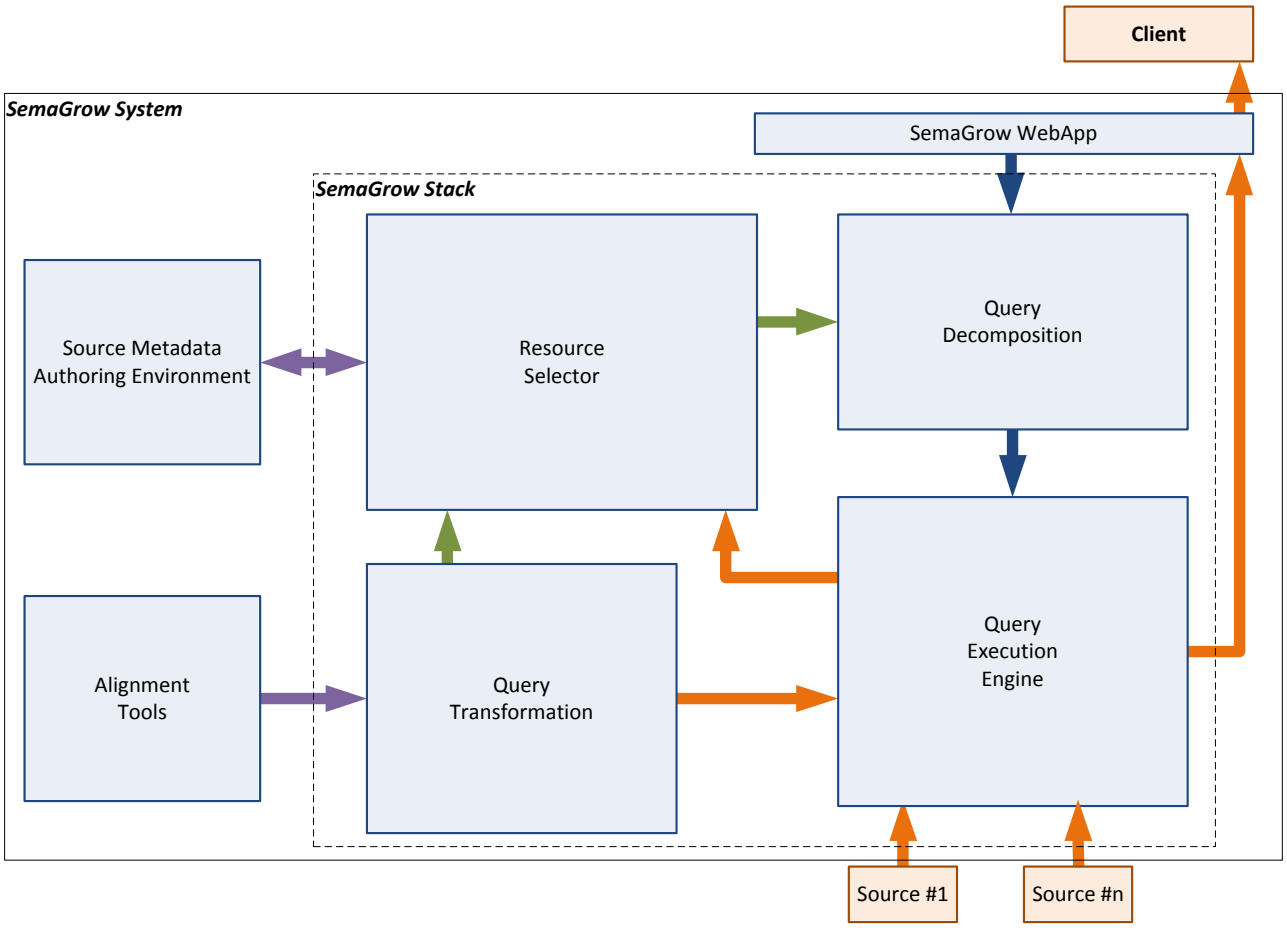

Figure 1: SemaGrow System Architecture

Furthermore, we established and ran experiments to evaluate our methods for managing heterogeneity:

- We used the heterogeneous data sources from the Reactive Resource Discovery use cases to measure the validity of ontology alignment, by comparing automatically calculated mappings against prior manual mappings.
- We used the FAO document base to evaluate the appropriateness of the PEARL language and the CODA architecture to cover a broad variety of extraction/classification and triplification scenarios.

Finally, we developed the *RDF triple generator of realistic datasets*, which will be used during the final period of the project to extrapolate SemaGrow tests on projected future data sizes.

4. Prototyping and Deployment

In order to be able to deploy and test the SemaGrow system, the project prepared and maintains hardware infrastructure, integrated the research prototypes described previously in a SemaGrow Stack prototype, and developed the various software that comprises the SemaGrow ecosystem of tools, including some peripheral and domain-specific tools that are not part of the core SemaGrow system.

4.1 The SemaGrow System

The SemaGrow System (Figure 1) comprises:

- The SemaGrow Stack, which integrates the research prototypes described in Section 3.1 under the Sesame architecture (http://rdf4j.org). Integration has advanced to the point of having a functional prototype, with some remaining issues pertaining to the persistency and importing of histograms to be tackled during the final project period. The SemaGrow Stack can be used from Java programs via the Sesame SAIL API. The

SemaGrow Stack is developed on Github, http://github.com/semagrow and is also distributed in binary packages for Ubuntu Lucid from the SWC repository http://semagrow.semantic-web.at/deb/

- The SemaGrow WebApp accesses SemaGrow Stack functionality via the SAIL API to expose a SPARQL endpoint for remote clients and a Web user interface for human querying of a SemaGrow Stack deployment. The SemaGrow WebApp is also developed on http://github.com/semagrow and distributed from the SWC .deb repository. A demo deployment is available at http://semagrow.eu/?q=service
- The Automated Rigorous Tester (ART), which automates the execution of the SemaGrow test suites (cf. Section 3.3) and generate system health reports. ART is still in an early stage of development: although experimental setup development has advanced, the bulk of the implementation and integration work is scheduled for the final project period.
- The alignment tools (cf. Section 3.2) used to provide the vocabulary mappings upon which the SemaGrow Stack relies to handle heterogeneity. Alternative alignment tools can be used, as long as a list of mappings of URI resources is produced. An import facility for third-party mappings is under development.
- The SemaGrow environment for visualizing and editing data source metadata. This is based on the ELEON Authoring Environment and is developed on Bitbucket.org, http://bitbucket.org/bigopendata/eleon Alternatively, any RDF editor can used to produce RDF following the Sevod schema. The guidelines included in the SemaGrow WebApp distribution explain how to import RDF metadata in a SemaGrow deployment.

These components make up the generic SemaGrow System prototype, not tied to any of the three demonstrators developed for the SemaGrow use cases.

4.2 SemaGrow Deployments

The project develops three demonstrators, that is, three SemaGrow Stack client applications, which will be used to validate the three use case categories presented in Section 2:

The *Trees4Future/AgMIP Demonstrator* validates the *Semagrow* system on the *Heterogeneous Data Collections & Streams* use cases. This demonstrator focuses on the support of scientific communities in the domain of agricultural and forestry modelling. Specifically, it supports end-users in preparing suitable input dataset for their modelling exercises from the wealth of heterogeneous big data collections and streams available, validating the integrative semantic capabilities provided by the *Semagrow Stack* as well as its ability to search and retrieve large data volumes.

The *AGRIS Demonstrator* validates the *Semagrow* system on the *Reactive Data Analysis* use cases. The *AGRIS Web Portal* serves bibliographical data from the agricultural domain, combined with widgets that retrieve and present Web resources that are relevant to the currently viewed bibliography item. The AGRIS Demonstrator validates that using the *Semagrow Stack* as a backend supports IT personnel more easily extend the range of data sources federates under the AGRIS Web Portal; and that the *Semagrow Stack* is able to reactively search through big data in order to find results that are not voluminous.

The *Agricultural Discovery Space (ADS)* application validates the *Semagrow* system on the *Reactive Resource Discovery* use cases. ADS provides the user with educational material that can be used in teaching and training activities, including academic publications, lectures, and presentations. The ADS Demonstrator validates Semagrow technologies on Web developers with simpler, NoSQL querying use cases and for use cases where end-users need to search a large number of data sources, although the contents of any individual data source do not constitute big data by itself.

In order to achieve these deployments, the project prepares and develops hardware infrastructure and software for accessing it, server-side tools, clients and client-side tools, seeding an *ecosystem* of software tools, semantic vocabularies, and public Web services developed around the main SemaGrow system. Specifically:

- Client-side user interfaces for accessing data federated under a SemaGrow Stack deployment: the Trees4Future/AgMIP GUI for preparing agro-environmental modelling experiments, the AGRIS Web portal, and ADS.
- A variety of triplifiers of data in the NetCDF model, data in XML schemas commonly found in the agricultural domain and in publishing educational resources, and other formats and data models from the SemaGrow application domain.
- Where appropriate RDF schemas have not been defined, the project developed them. It is expected that some can be of general value, such as the RDF schema derived from the NetCDF data model. Our netCDF schema extends the RDF Data Cube Vocabulary (http://www.w3.org/TR/vocab-data-cube), the W3C Recommendation for representing science data. This is on-going work and the vocabulary has not been published yet, but will be published during the final project period.

- Collected and organized taxonomies and ontologies that can be used to represent in RDF the values in commonly used codelists and standard vocabularies used in climate and forecast NetCDF datasets. This includes, among others, the Climate and Forecasting conventions (http://cfconventions.org), the scientific measurements units used in the ISI-MIP repository and the ICASA-list of variables used as a taxonomy with AgMIP datasets.
- The Toolkit for Repository Integration, tools for using large-scale distributed computational infrastructure to apply the triplifiers above to extract RDF data from non-RDF datasets and to expose the resulting RDF data via SPARQL endpoints.
- The rdfCDF Toolkit, comprising tools for converting and serving NetCDF data as well as for preparing NetCDF files by consuming data exposed on SPARQL endpoints. rdfCDF integrates the NetCDF triplifier (developed previously) with the NetCDF Creator and the NetCDF Endpoint. The NetCDF Endpoint satisfies the unforeseen piloting requirement that no triplified duplicates of large-scale NetCDF collections should be maintained. The NetCDF Endpoint exposes raw NetCDF data through a SPARQL endpoint, by translating between SPARQL and the NetCDF Java Library (http://www.unidata.ucar.edu/software/thredds/current/netcdf-java), an open-source implementation of the Common Data Model that underlies NetCDF, OpenDAP, and HDF5.
- The project integrated Web crawling and semantic annotation tools into a system that crawls the Web for resources relevant to agriculture, indexes them with AGROVOC terms, and exposes the index as a SPARQL endpoint that can be joined with AGRIS to look for Web resources that are thematically related to a given AGRIS bibliographic entry.
- The project provides cleanDT, an endpoint that serves a structured and well-formed publication date for AGRIS bibliography entries. The dataset is automatically constructed by applying heuristics to the (often) informal values used in the AGRIS publication date field. These dates can be more accurately joined against temporal specifications in other datasets used in the Reactive Data Analysis pilots.
- The project developed SemagrowREST, a REST API that wraps the Semagrow Stack WebApp under a NoSQL querying endpoint. This layer reduces the flexibility of what can be queried, but simplifies client development for those cases that it does support.

These SemaGrow deployments have validated the ability to deploy SemaGrow with minimal disruption of existing workflows, while also gauging end-user satisfaction with the new features and capabilities. Furthermore, SemaGrow pilots have aimed to gauge ease of use for IT personnel who install and maintain it and for programmers who develop client software. In these pilots, besides the SemaGrow software tools we are also testing our *guidelines* and *getting started* document distributed together with the SemaGrow Stack and WebApp. The installation process and the guides have been refined through multiple iterations of making SemaGrow available to *Hackathon* participants: consortium members providing support for the Hachathon noted the questions and calls for support and carried out the appropriate updates to simplify a procedure or clarify a point in the documentation.

5. Steps to Deliver Impact

With respect to scientific publications, the consortium produced one (1) journal article and seven (7) conference papers discussing SemaGrow technologies, research outcomes, and impact to the SemaGrow end-users communities. Two (2) more conference papers have been prepared and submitted, decision pending at the time of writing this report. Besides academic dissemination, the project has also identified potential users of SemaGrow technologies and has worked out a strategy for disseminating to developer and user communities that are to be engaged into building on SemaGrow technology beyond project's end.

As part of this strategy, the project organized two *Hackathons* to engage developers in using SemaGrow technology and to obtain valuable feedback from practical use of offered data and methods.

6. Contact Information

Project website: www.semagrow.eu

Project Coordinator: Prof. Miguel A. Sicilia
Universidad de Alcalá
msicilia@uah.es
http://www.cc.uah.es/msicilia/

Scientific Manager: Dr Vangelis Karkaletsis
National Centre for Scientific Research "Demokritos"
vangelis@iit.demokritos.gr
http://users.iit.demokritos.gr/~vangelis/

Technical Manager: Dr Stasinos Konstantopoulos
National Centre for Scientific Research "Demokritos"
konstant@iit.demokritos.gr
http://www.iit.demokritos.gr/people/konstantopoulos-stasinos

Universidad de Alcalá
Spain

NCSR "Demokritos"
Greece

Università di Tor Vergata
Italy

Semantic Web Company
Austria

Institute of Physics Belgrade
Serbia

Stichting Dienst Landbouwkundig Onderzoek
The Netherlands

Food and Agriculture Organization of the UN
Italy

Agro-Know Technologies
Greece